Importing Libraries

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import warnings
import seaborn as sns
```

In [2]:

```
%matplotlib inline
warnings.filterwarnings('ignore')
```

Loading the Dataset

In [3]:

```
df = pd.read_csv('student.csv')
```

First 5 rows of dataframe

In [4]:

```
df.head()
```

Out[4]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

Data Summary

In [5]:
```
df.describe()
```

Out[5]:

	math score	reading score	writing score
count	1000.00000	1000.000000	1000.000000
mean	66.08900	69.169000	68.054000
std	15.16308	14.600192	15.195657
min	0.00000	17.000000	10.000000
25%	57.00000	59.000000	57.750000
50%	66.00000	70.000000	69.000000
75%	77.00000	79.000000	79.000000
max	100.00000	100.000000	100.000000

Datatypes in Dataframe

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 8 columns):
 #   Column                       Non-Null Count  Dtype 
---  ------                       --------------  ----- 
 0   gender                       1000 non-null   object
 1   race/ethnicity               1000 non-null   object
 2   parental level of education  1000 non-null   object
 3   lunch                        1000 non-null   object
 4   test preparation course      1000 non-null   object
 5   math score                   1000 non-null   int64 
 6   reading score                1000 non-null   int64 
 7   writing score                1000 non-null   int64 
dtypes: int64(3), object(5)
memory usage: 62.6+ KB
```

Column Names

In [7]:
```
df.columns
```

Out[7]:
```
Index(['gender', 'race/ethnicity', 'parental level of education', 'lunch',
       'test preparation course', 'math score', 'reading score',
       'writing score'],
      dtype='object')
```

Numerical Columns

```
In [8]: df.columns[df.dtypes=='int'].to_list()
```

```
Out[8]: ['math score', 'reading score', 'writing score']
```

Object Columns

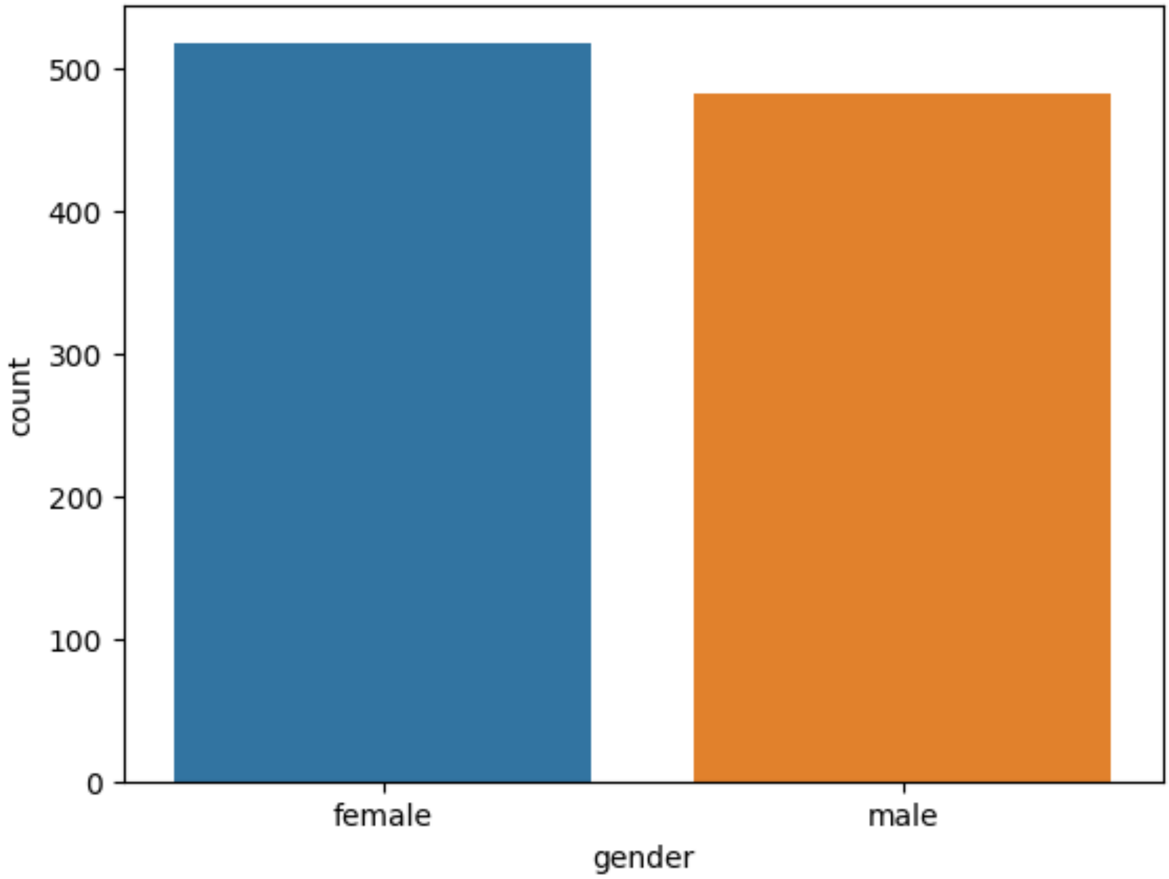
```
In [9]: df.columns[df.dtypes=='object'].to_list()
```

```
Out[9]: ['gender',
         'race/ethnicity',
         'parental level of education',
         'lunch',
         'test preparation course']
```

Count-Plot for gender

```
In [10]: sns.countplot(data=df, x='gender')
```

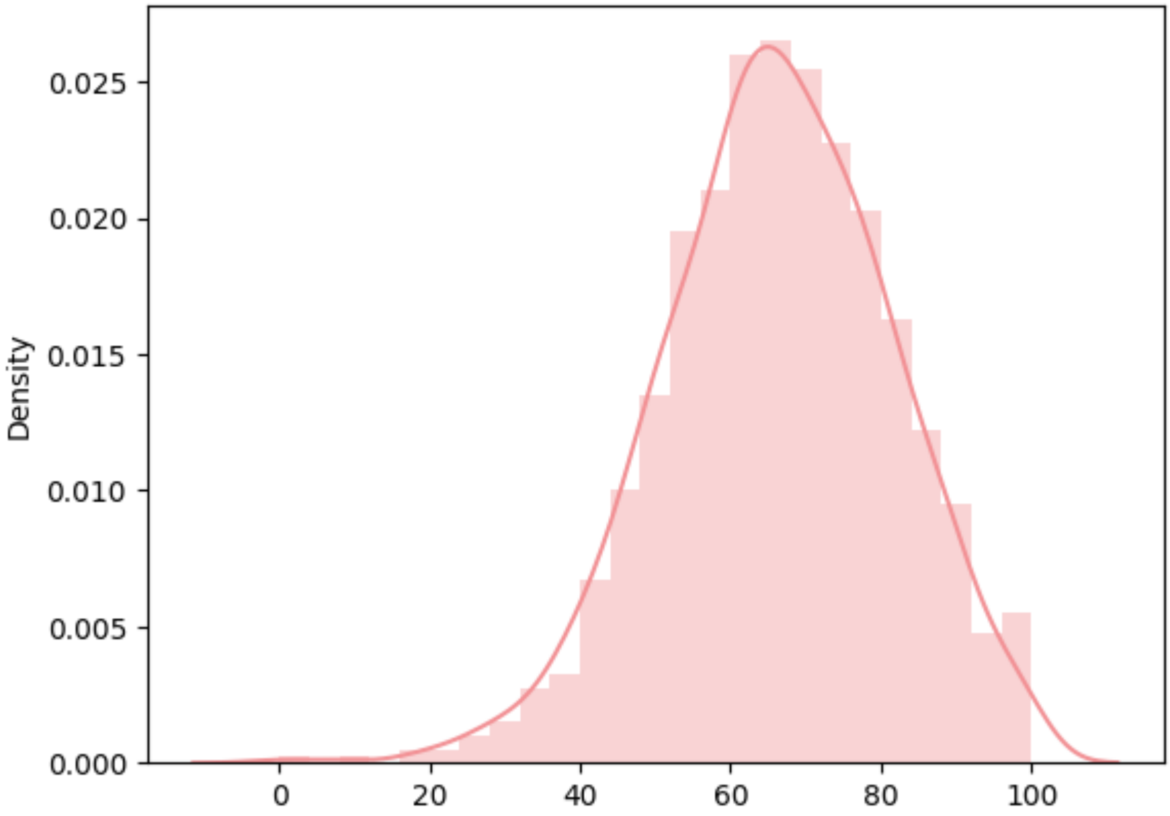
```
Out[10]: <AxesSubplot:xlabel='gender', ylabel='count'>
```



Distribution of Maths Score

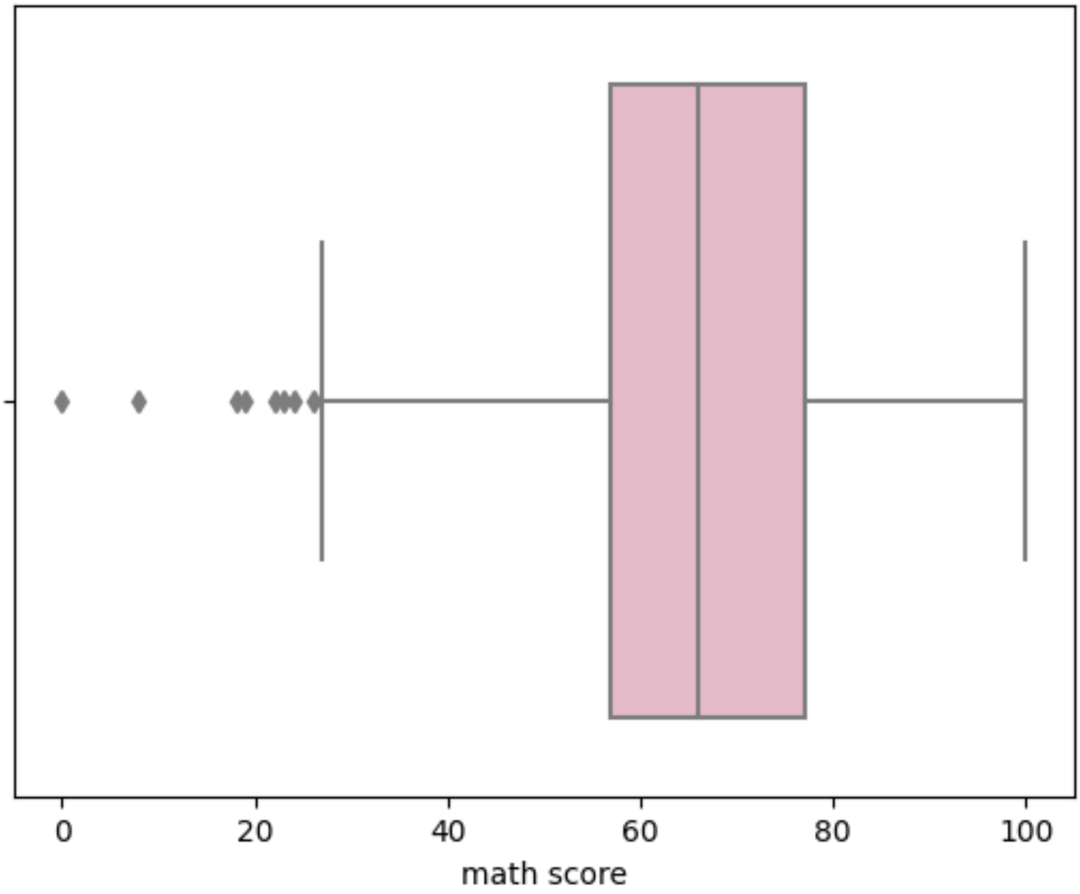
```
In [11]: sns.distplot(x = df['math score'], color = '#f29195')
```

```
Out[11]: <AxesSubplot:ylabel='Density'>
```



```
In [12]: sns.boxplot(x = 'math score', data=df, color='#edb4c6')
```

```
Out[12]: <AxesSubplot:xlabel='math score'>
```

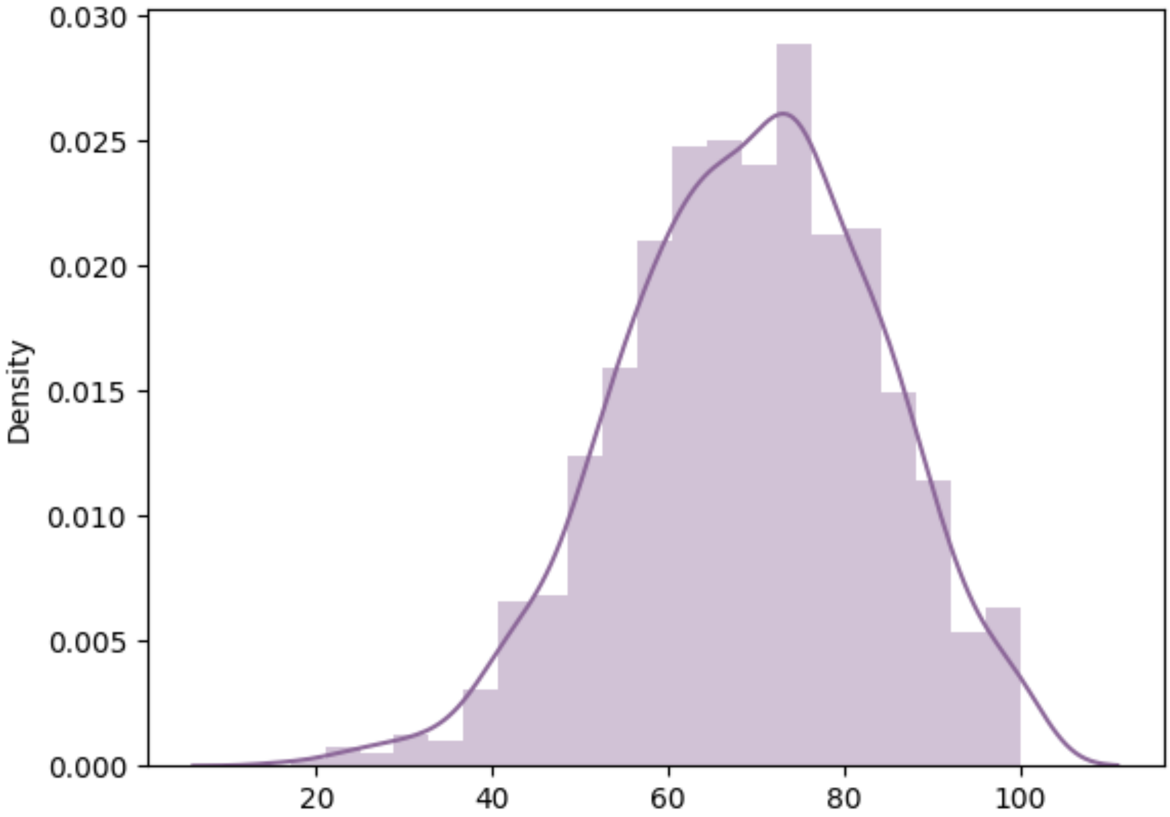


Observation : The distribution for Maths Score in the dataset is little left-skewed hence some of the students has performed poor.

Distribution of Reading Score

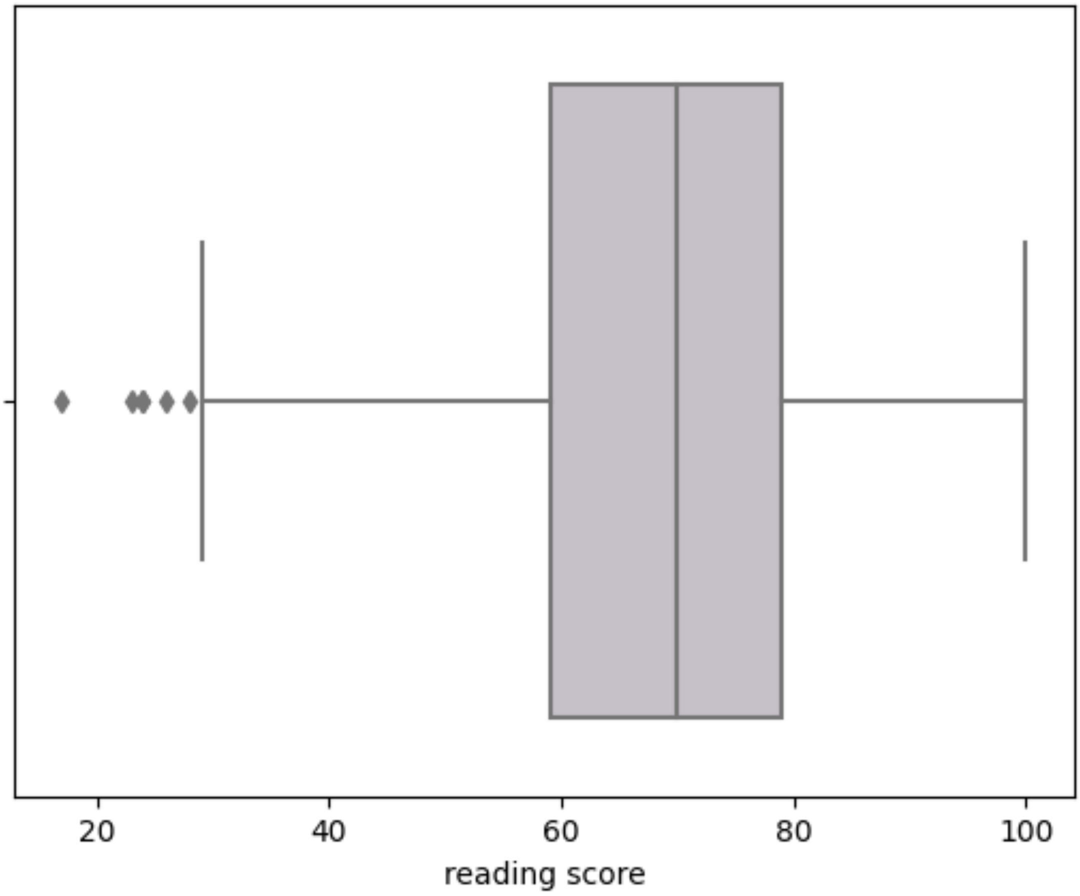
```
In [13]: sns.distplot(x = df['reading score'], color = '#8c6799')
```

```
Out[13]: <AxesSubplot:ylabel='Density'>
```



In [14]:
```
sns.boxplot(x = 'reading score', data=df, color="#c7bfc9")
```

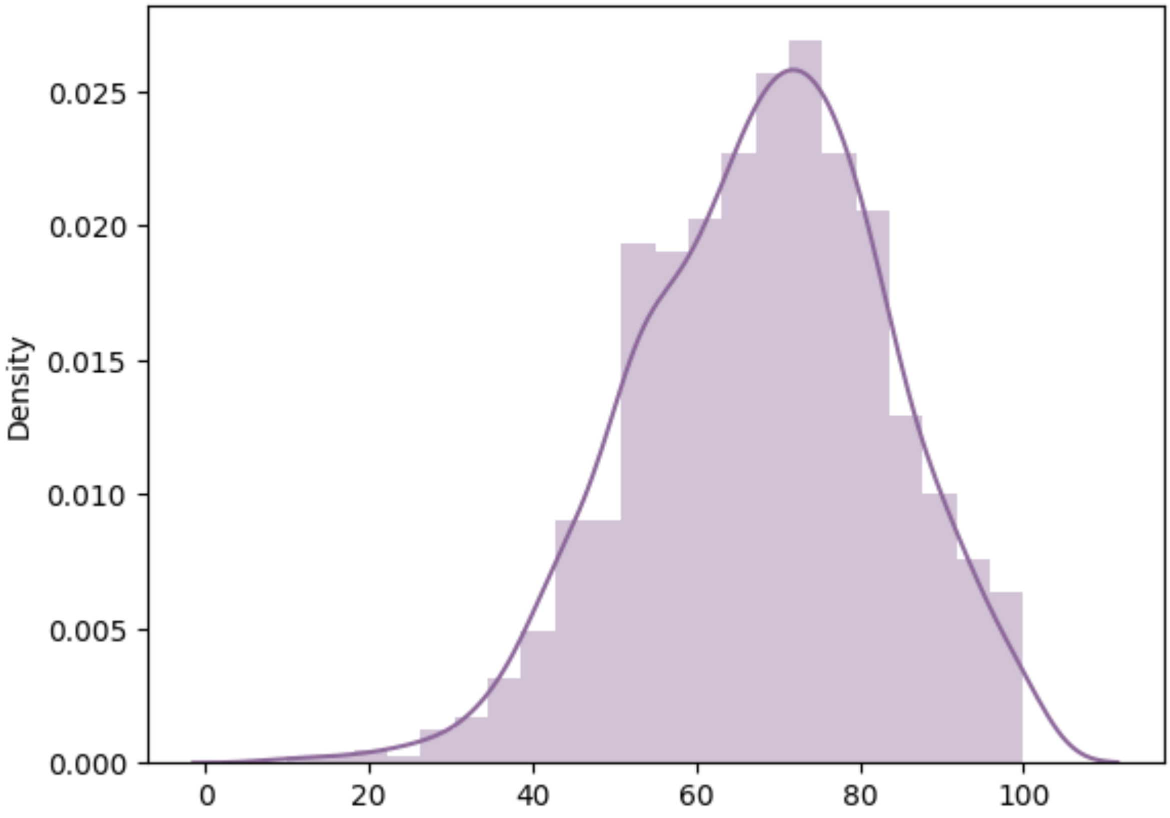
Out[14]:
```
<AxesSubplot:xlabel='reading score'>
```



Distribution of Writing Score

```
In [15]: sns.distplot(x = df['writing score'], color = '#8c6799')

Out[15]: <AxesSubplot:ylabel='Density'>
```

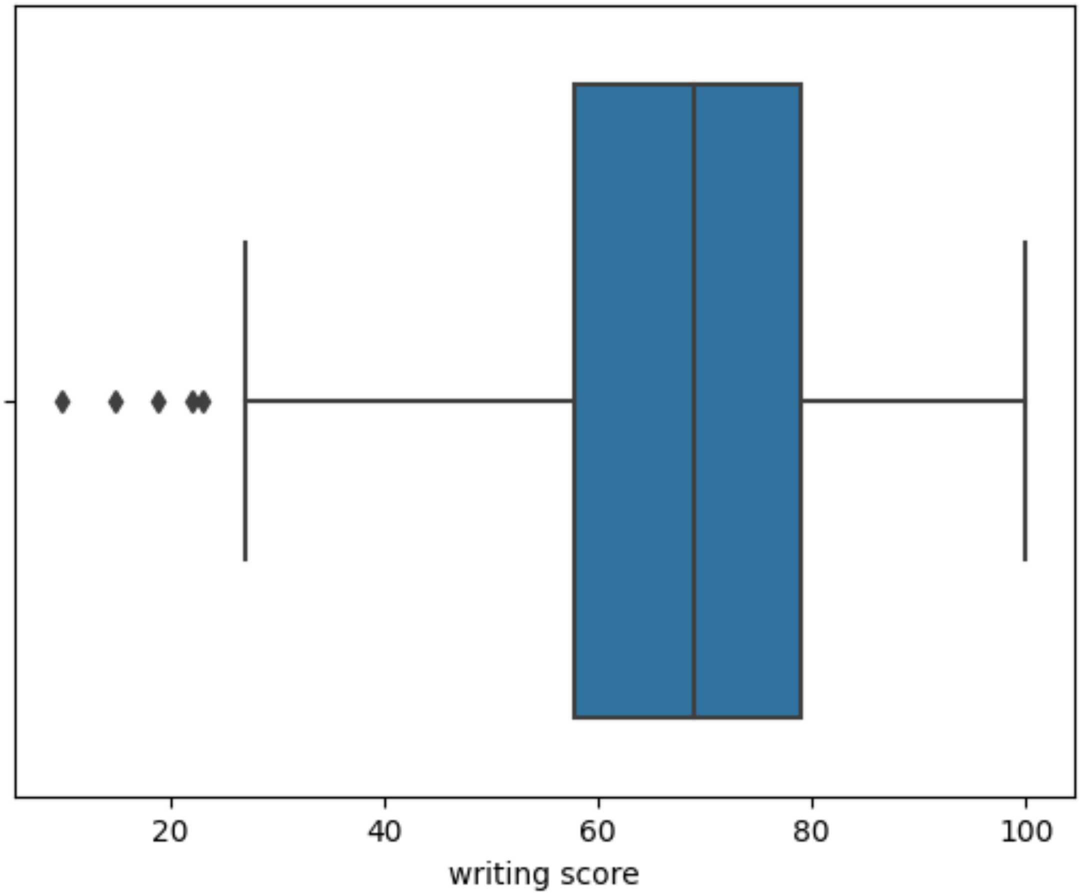


In [16]:

```
sns.boxplot(x = 'writing score', data=df)
```

Out[16]:

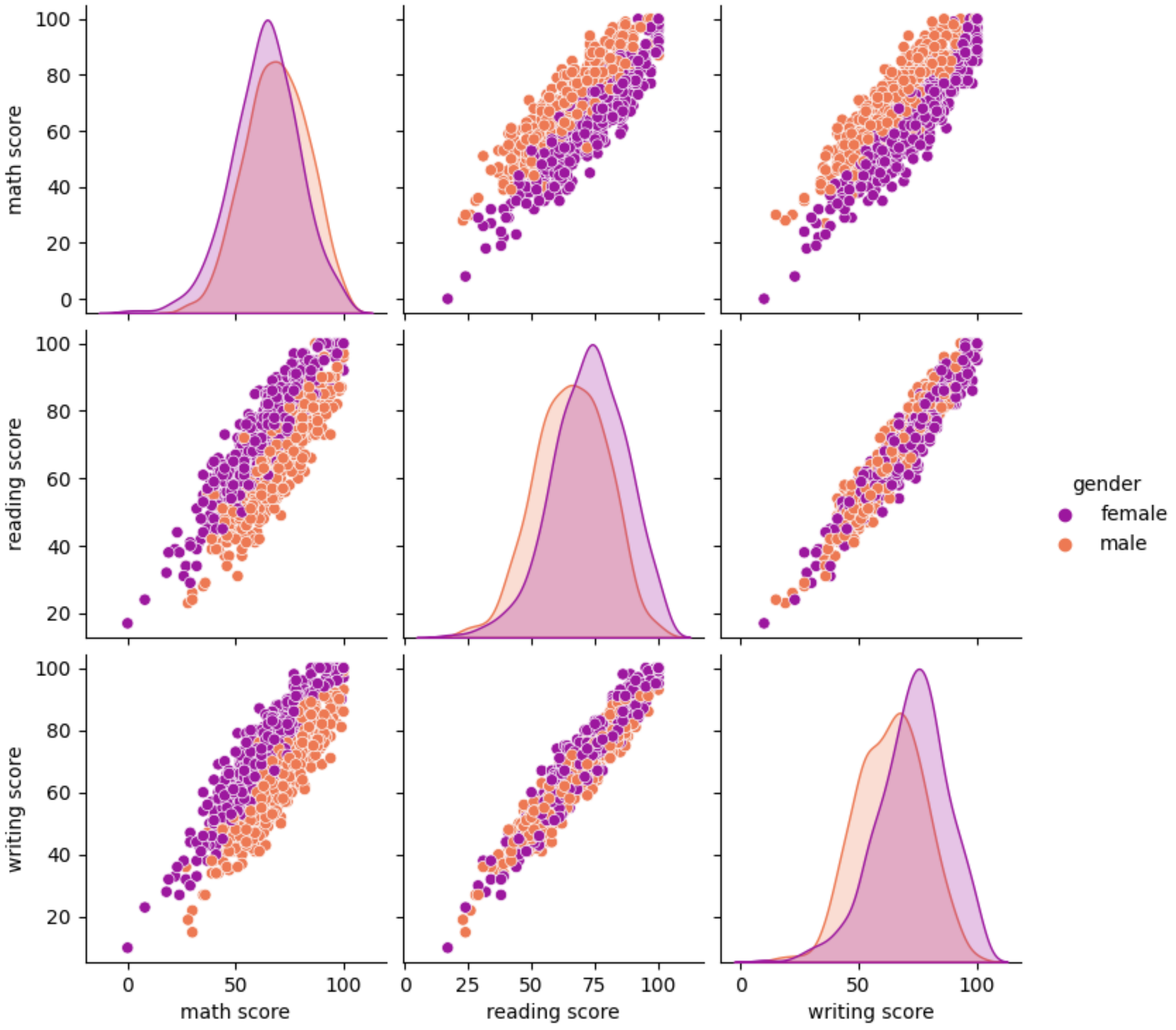
```
<AxesSubplot:xlabel='writing score'>
```



Pairplot for numerical features in the dataset wrt Gender

In [17]:
```
sns.pairplot(data=df, diag_kind="kde", palette="plasma", hue='gende
```

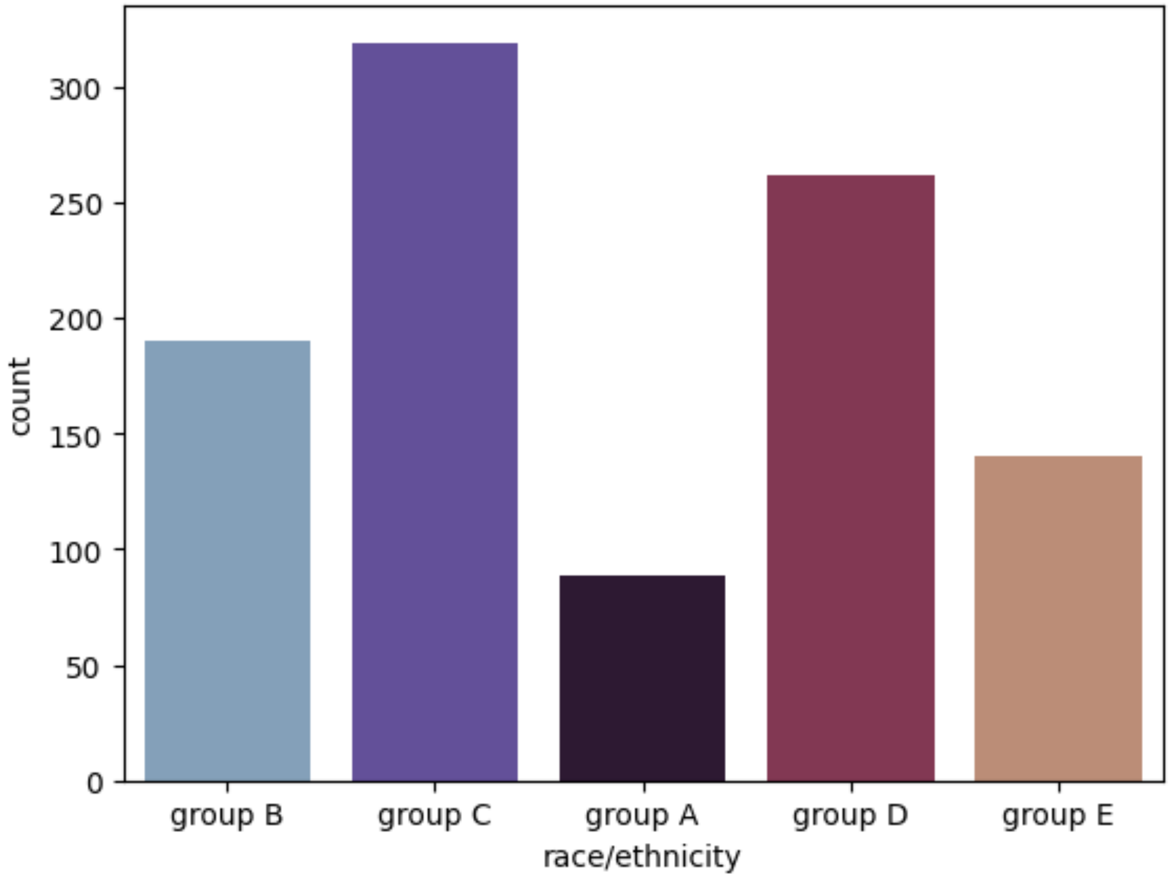
Out[17]:
```
<seaborn.axisgrid.PairGrid at 0x156c8cb50>
```



Count Plot for Race/Ethinicity

```
In [18]: sns.countplot(data=df, x='race/ethnicity', palette="twilight")
```

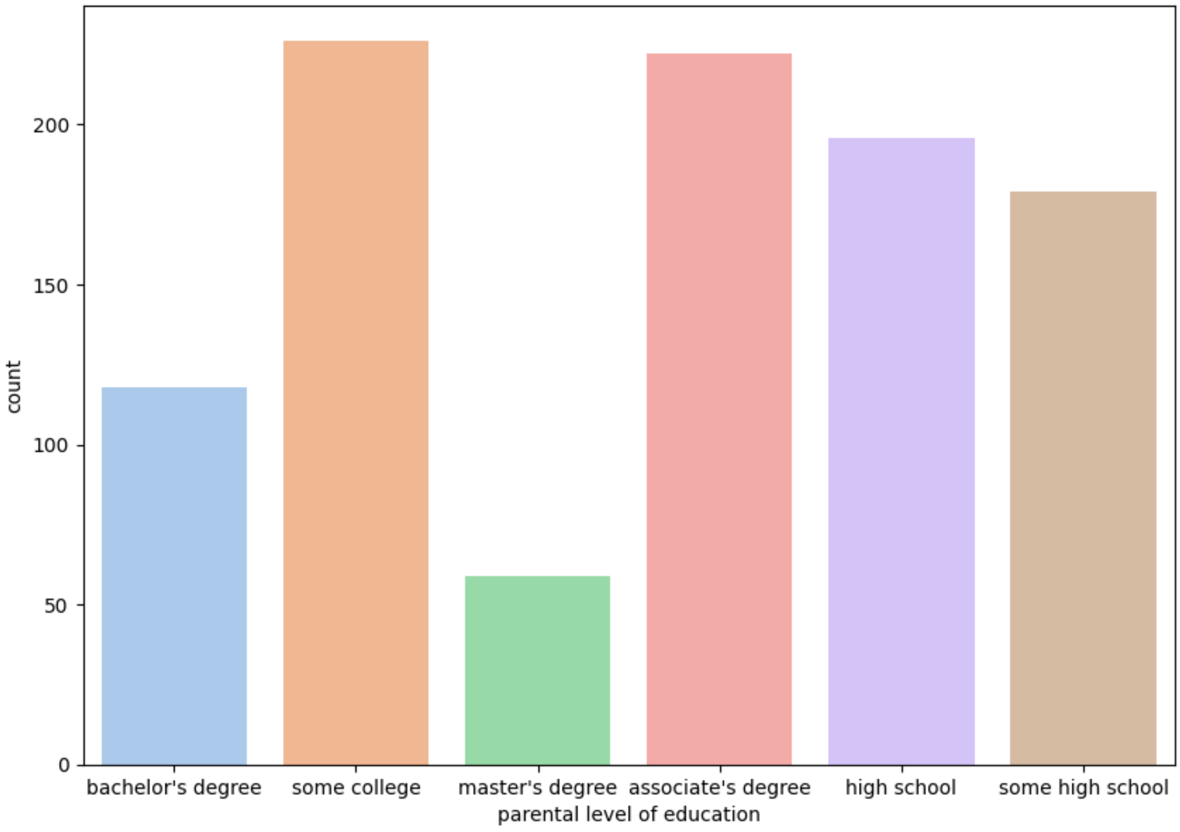
```
Out[18]: <AxesSubplot:xlabel='race/ethnicity', ylabel='count'>
```



Count Plot for Parental Education

In [19]:
```
plt.figure(figsize=(10, 7))
sns.countplot(data=df, x='parental level of education', palette="pa
```

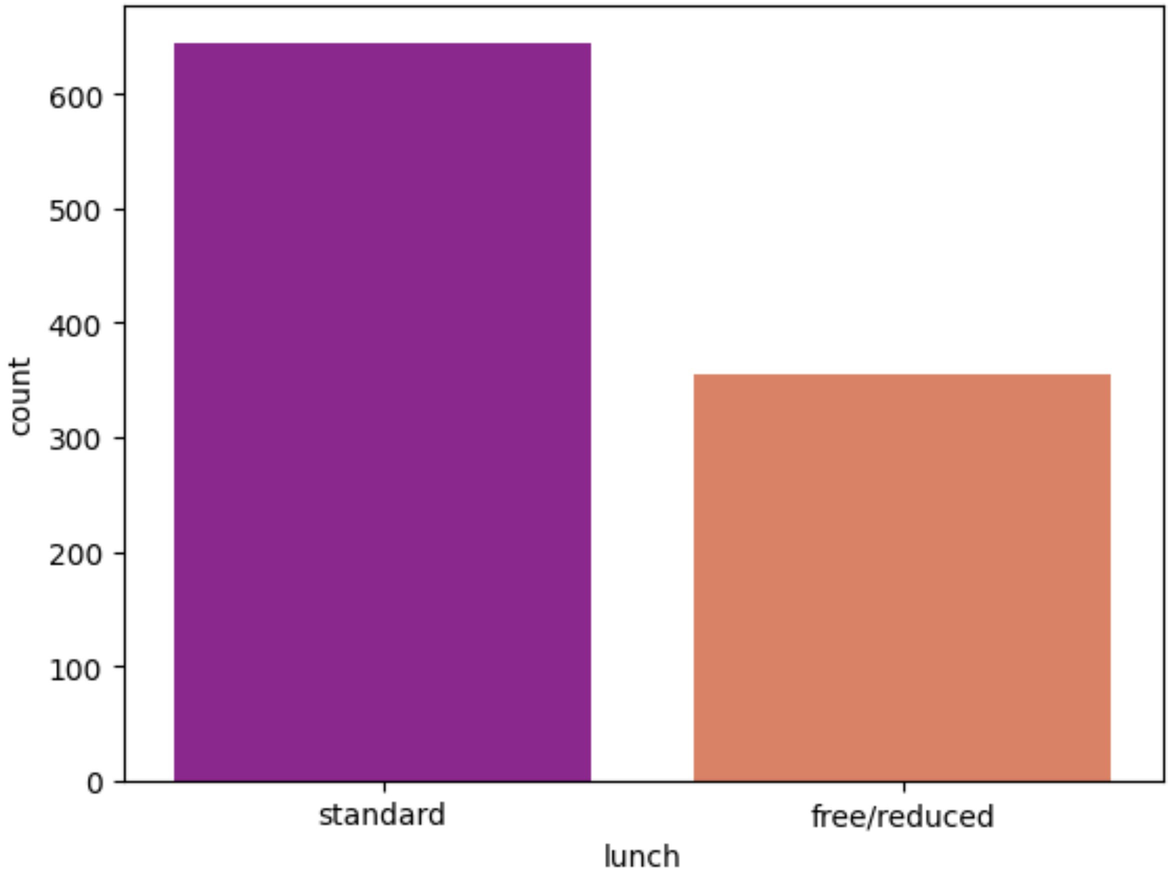
Out[19]:
```
<AxesSubplot:xlabel='parental level of education', ylabel='count'>
```



Count Plot for Lunch

In [20]:
```
sns.countplot(data=df, x='lunch', palette="plasma", lw=4)
```

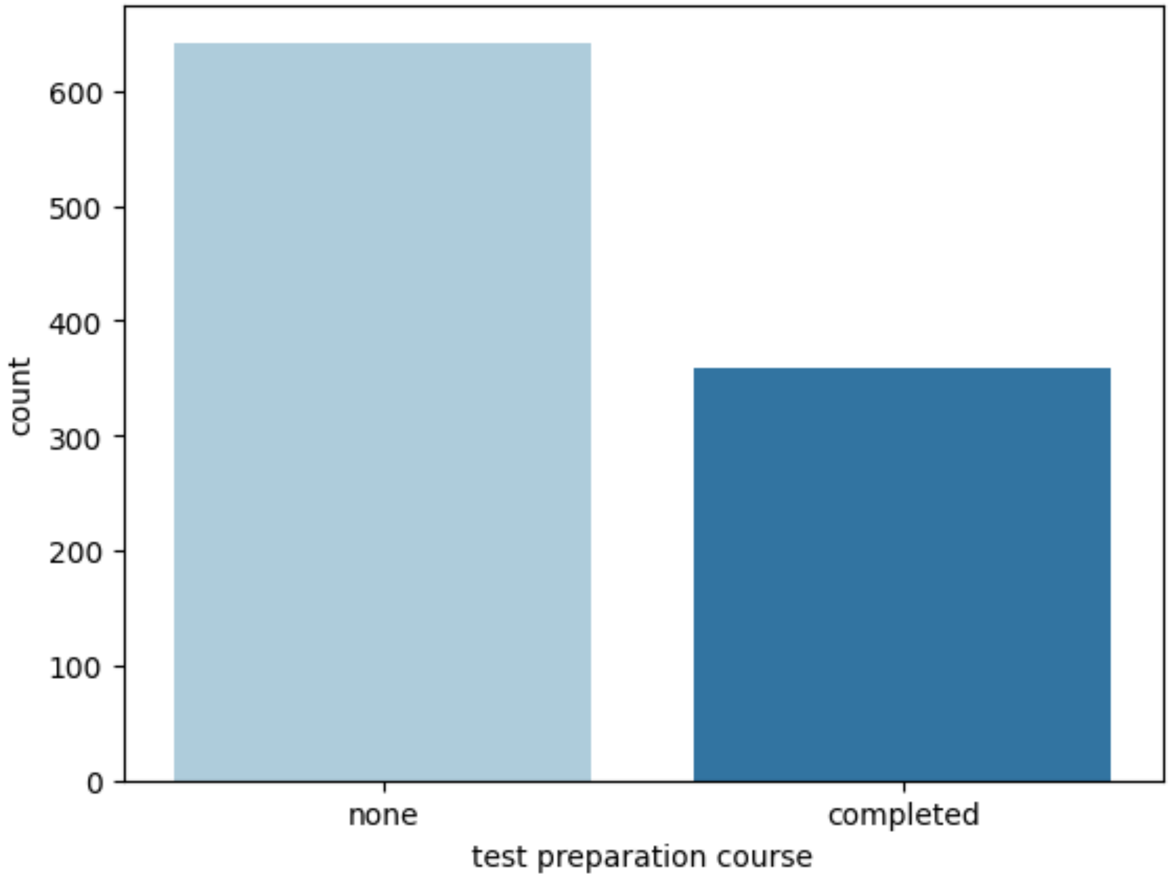
Out[20]:
```
<AxesSubplot:xlabel='lunch', ylabel='count'>
```



Count Plot for Test Preparation Course

In [21]:
```
sns.countplot(data=df, x='test preparation course', palette="Paired
```

Out[21]:
```
<AxesSubplot:xlabel='test preparation course', ylabel='count'>
```



Creation of new column - Average Score

In [22]:
```
df['Average Score'] = df[['math score', 'reading score',
         'writing score']].sum(axis=1)/3
```

First 5 Lines of dataset

In [23]:
```
df.head()
```

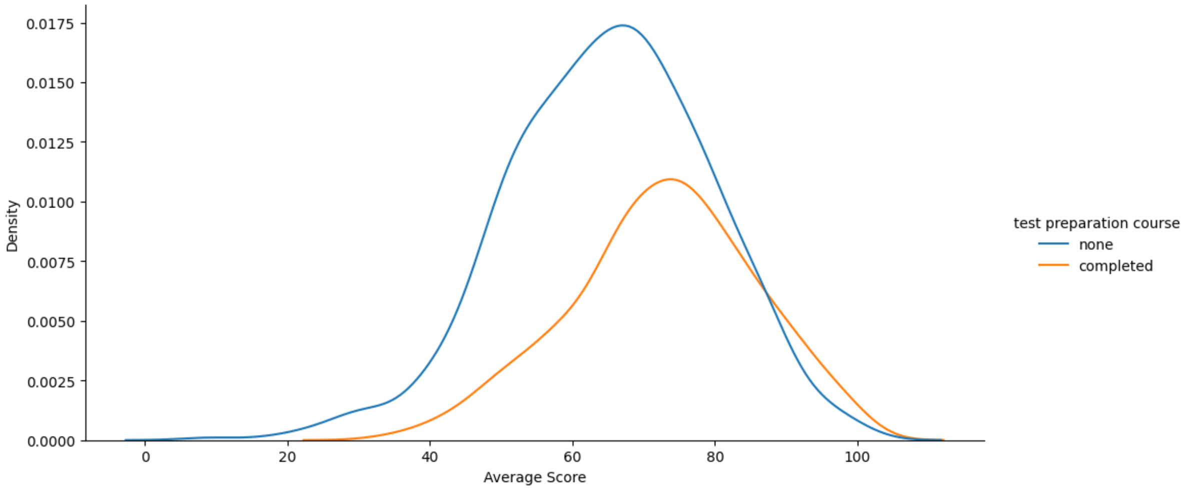
Out[23]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score	A
0	female	group B	bachelor's degree	standard	none	72	72	74	72.
1	female	group C	some college	standard	completed	69	90	88	82.
2	female	group B	master's degree	standard	none	90	95	93	92.
3	male	group A	associate's degree	free/reduced	none	47	57	44	49.
4	male	group C	some college	standard	none	76	78	75	76.

Test preapration and Average Score analysis

In [24]:
```
sns.displot(data = df, x = "Average Score", hue = 'test preparation
```

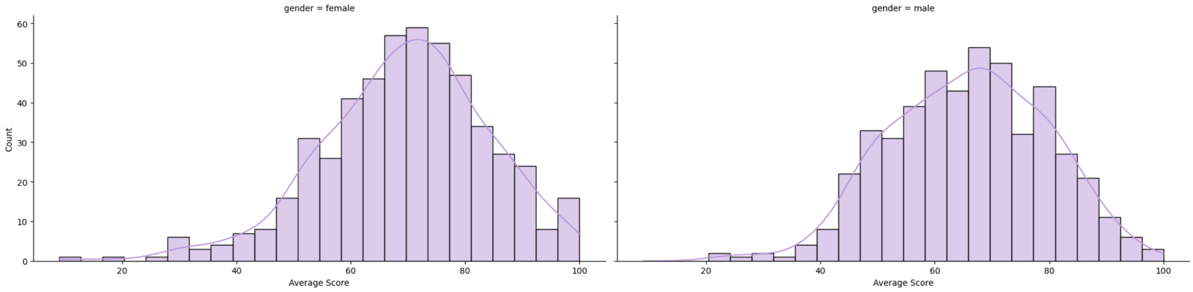
Out[24]:
```
<seaborn.axisgrid.FacetGrid at 0x15725d4e0>
```



Average Score wrt Gender

```
In [25]: sns.displot(data = df, x = "Average Score", col = 'gender', aspect=
```

```
Out[25]: <seaborn.axisgrid.FacetGrid at 0x1572b7ca0>
```



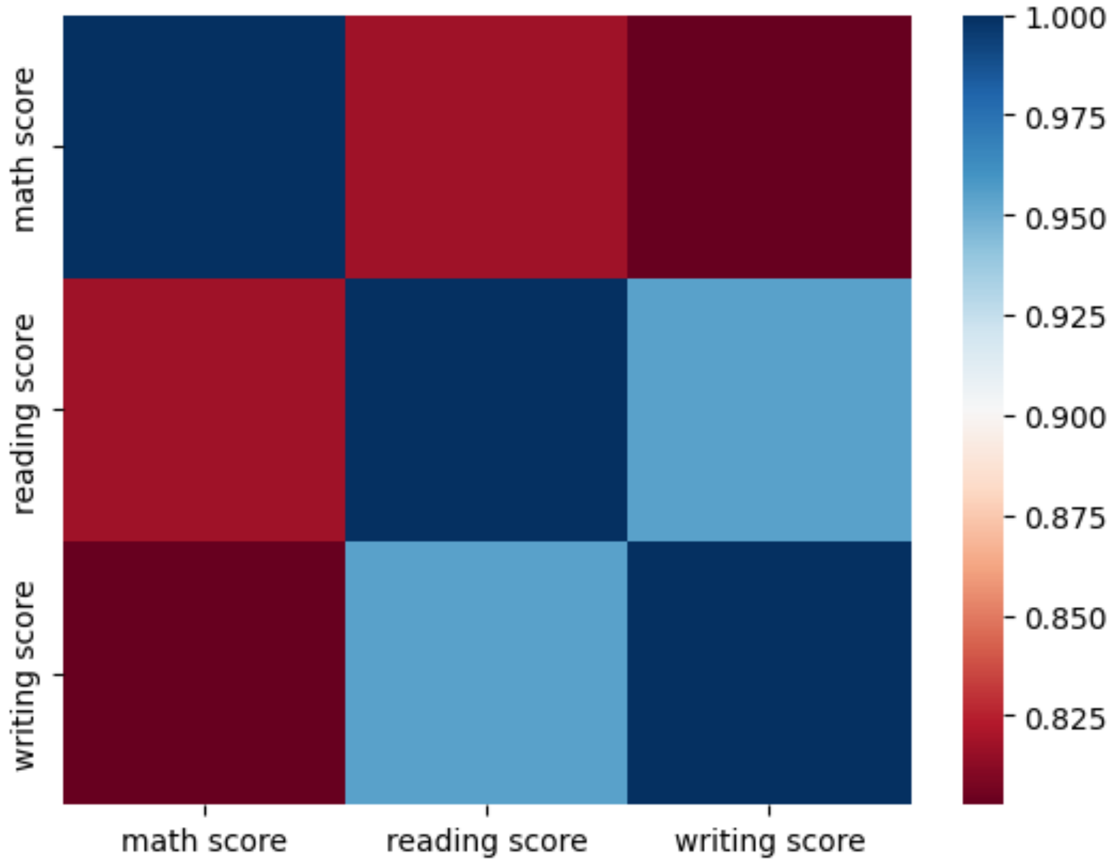
Correlation HeatMap

In [26]:

```
scores = ['math score', 'reading score',
          'writing score']
sns.heatmap(df[scores].corr(), cmap="RdBu")
df[scores].corr()
```

Out[26]:

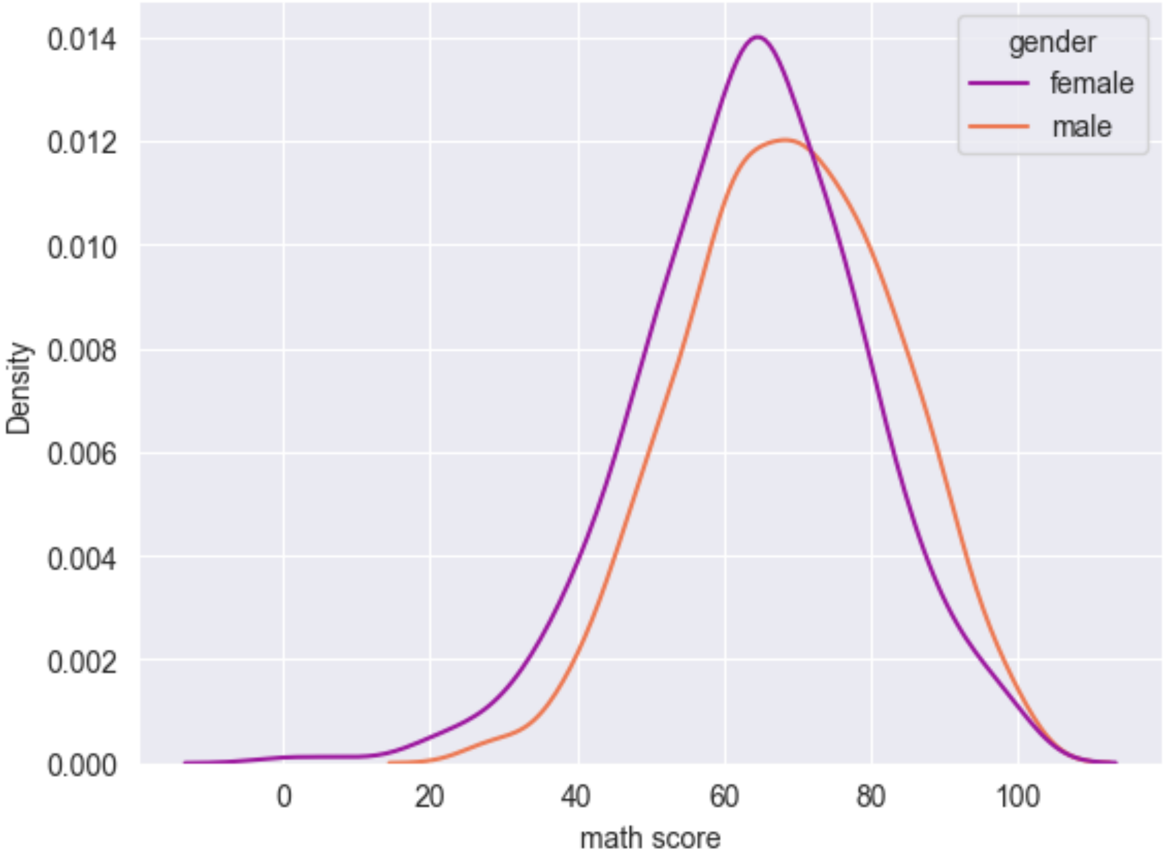
	math score	reading score	writing score
math score	1.000000	0.817580	0.802642
reading score	0.817580	1.000000	0.954598
writing score	0.802642	0.954598	1.000000



Maths Score wrt Gender

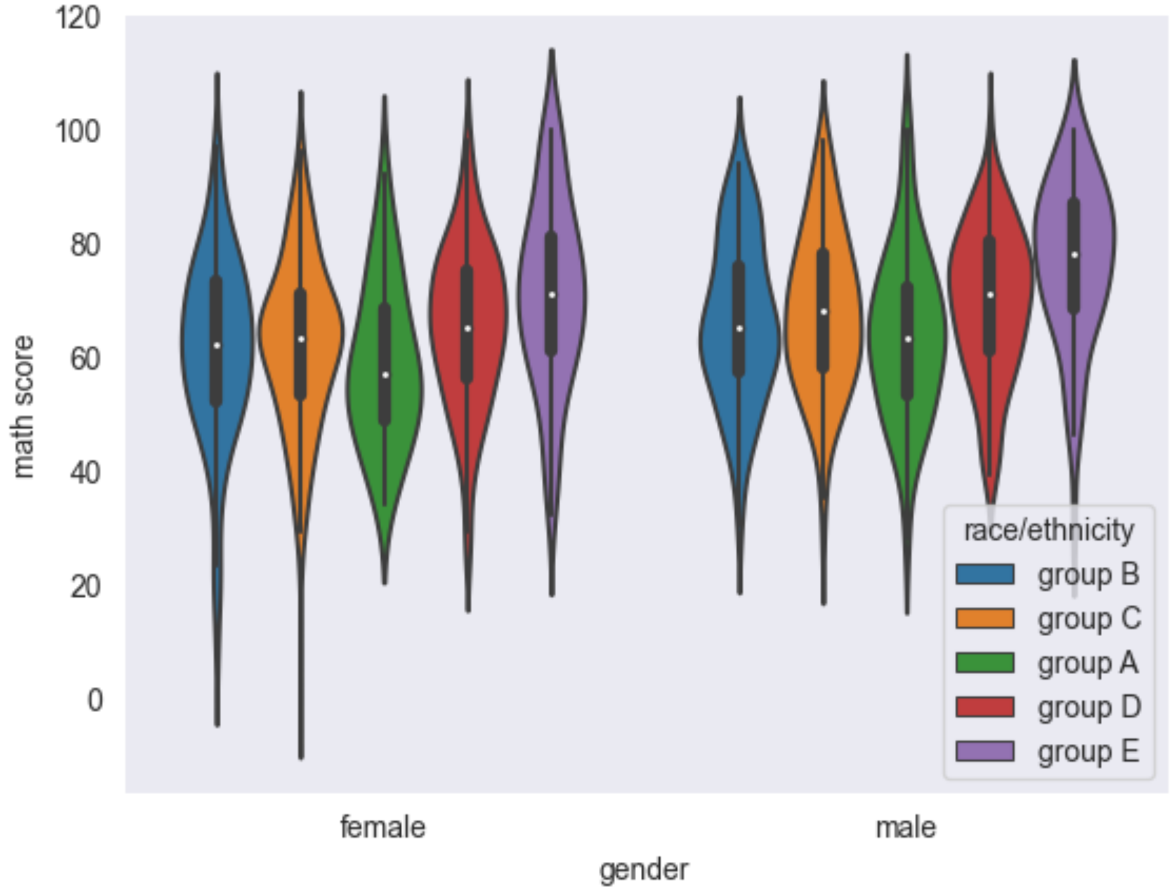
```
In [27]: sns.set_style('darkgrid')
         sns.kdeplot(x = df['math score'], hue=df['gender'], palette="plasma
```

```
Out[27]: <AxesSubplot:xlabel='math score', ylabel='Density'>
```



```
In [28]: sns.set_style('dark')
         sns.violinplot(data = df, y = "math score", x="gender", hue='race/e
```

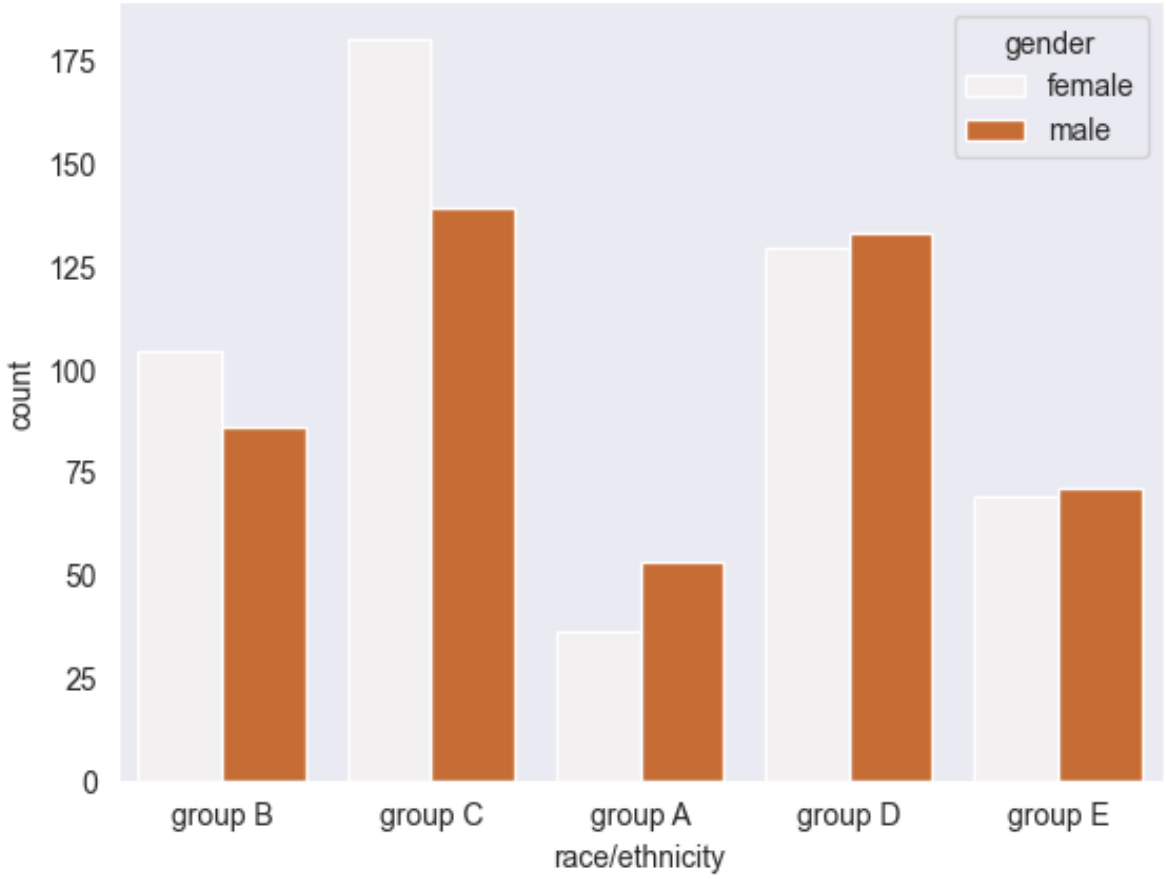
```
Out[28]: <AxesSubplot:xlabel='gender', ylabel='math score'>
```



Count of Race wrt to gender

In [29]:
```
sns.set_style('dark')
sns.countplot(data=df, x = 'race/ethnicity', hue="gender", color="#
```

Out[29]:
```
<AxesSubplot:xlabel='race/ethnicity', ylabel='count'>
```



In [ ]: